COMPUTER SCIENCE TRIPOS Part IB

Tuesday 5 June 2001 1.30 to 4.30

Paper 4

Answer **five** *questions.*

Submit the answers in five **separate** *bundles, each with its own cover sheet. On each cover sheet, write the numbers of* **all** *attempted questions, and circle the number of the question attached.*

Write on **one** *side of the paper only.*

You may not start to read the questions printed on the subsequent pages of this question paper until instructed that you may do so by the Invigilator

1 Continuous Mathematics

The complex form of the Fourier series is:

$$f(x) = \sum_{k=-\infty}^{+\infty} c_k e^{i2\pi kx}$$

where c_k is a complex number and $c_k = c^*_{-k}$.

(*a*) Prove that the complex coefficient, c_k, encodes the amplitude and phase coefficients, A_k and ϕ_k, in the alternative form:

$$f(x) = \sum_{k=0}^{+\infty} A_k \cos(2\pi kx - \phi_k)$$

[10 marks]

(*b*) What is special about the case $k = 0$? [2 marks]

(*c*) Explain how the coefficients, c_k, of the Fourier series of the periodic function, $f(x)$:

$$f(x) = f(x + T), \forall x$$

can be obtained from the Fourier transform, $F_L(\nu)$, of the related function, $f_L(x)$:

$$f_L(x) = \begin{cases} f(x), & -\frac{T}{2} \leqslant x < \frac{T}{2} \\ 0, & \text{otherwise} \end{cases}$$

[8 marks]

2 Concurrent Systems

An interprocess communication environment is based on *synchronous* message passing. A server is to be designed to support a moderate number of simultaneous client requests.

Clients send a request message to the server, continue in parallel with server operation, then wait for the server's reply message.

Discuss the design of the server's interaction with the clients. Include any problems you foresee and discuss alternative solutions to them. [20 marks]

3 Further Java

(*a*) Describe how mutual-exclusion locks provided by the `synchronized` keyword can be used to control access to shared data structures. In particular you should be clear about the behaviour of concurrent invocations of different synchronized methods on the same object, or of the same synchronized method on different objects. [6 marks]

(*b*) Consider the following class definition:

```
class Example implements Runnable {
  public static Object o = new Object();
  int count = 0;

  public void run() {
    while (true) {
      synchronized (o) {
        count ++;
      }
    }
  }
}
```

Show how to start two threads, each executing this `run` method. [2 marks]

(*c*) When this program is executed, only one of the `count` fields is found to increment, even though threads are scheduled preemptively. Why might this be? [2 marks]

(*d*) Define a new class `FairLock`. Each instance should support two methods, `lock` and `unlock`, which acquire and release a mutual exclusion lock such that calls to `unlock` never block the caller, but will allow the longest-waiting blocked thread to acquire the lock. The lock should be *recursive*, meaning that the thread holding the lock may make multiple calls to `lock` without blocking. The lock is released only when a matched number of `unlock` operations have been made.

You may wish to make use of the fact the `Thread.currentThread()` returns the instance of `Thread` that is currently executing. [10 marks]

4 Compiler Construction

Consider the following grammar giving the concrete syntax of a language:

$$\begin{array}{lcl} E & \to & id \\ C & \to & E = E; \\ C & \to & \{B\} \\ C & \to & C\ \textbf{repeatwhile}\ E \\ C & \to & \textbf{if}\ E\ \textbf{then}\ C \\ C & \to & \textbf{if}\ E\ \textbf{then}\ C\ \textbf{else}\ C \\ B & \to & B\ C \\ B & \to & C \\ S & \to & C\ eof \end{array}$$

where C **repeatwhile** E has the same meaning as **do** C **while** E in C or Java.

(*a*) List the terminals and non-terminals of this grammar and explain the significance of S. [3 marks]

(*b*) Identify any ambiguities in the above grammar and rewrite it to remove them, ensuring that your new grammar generates exactly the same set of strings. [4 marks]

(*c*) Specify a suitable abstract syntax, for example by giving a type declaration in a programming language of your choice, which might be used to hold parse trees for this language. [3 marks]

(*d*) Give *either* a recursive descent parser *or* a characteristic finite state machine (e.g. for SLR(1)) with associated parser for your grammar. Your parser need not return a parse tree—it suffices for your parser either to accept or to reject the input string. [10 marks]

5 Data Structures and Algorithms

(*a*) Outline how you would determine whether the next line segment turns left or right during the Graham scan phase of the standard method of computing the convex hull of a set of points in a plane. [5 marks]

(*b*) Describe in detail an efficient algorithm to determine how often the substring `ABRACADABRA` occurs in a vector of 10^6 characters. Your algorithm should be as efficient as possible. [10 marks]

(*c*) Roughly estimate how many character comparisons would be made when your algorithm for part (*b*) is applied to a vector containing 10^6 characters uniformly distributed from the 26 letters `A` to `Z`. [5 marks]

6 ECAD

(*a*) When designing clocked circuits there are times when asynchronous inputs have to be sampled which may result in metastable behaviour in state holding elements. How might metastability be avoided when sampling asynchronous inputs? [5 marks]

(*b*) An optical shaft encoder (e.g. used on the internal rollers of a mechanical mouse) consists of a disk with an evenly spaced alternating transparent and opaque grating around the circumference. Two optical sensors are positioned such that when one sensor is at the middle of an opaque region, the other is at the edge. Consequently, the following Gray code sequence is produced, depending upon the direction of rotation:

positive rotation	negative rotation
00	00
01	10
11	11
10	01

time ↓

A shaft decoder module is required to convert the Gray code into an 8-bit position. The 8-bit position should be incremented every time the input changes from one state to another in a positive direction (e.g. from 00 to 01, or from 10 to 00). Similarly, the 8-bit position should be decremented every time the input changes from one state to another in a negative direction (e.g. from 00 to 10, or from 01 to 00).

Write and comment a Verilog module which performs the function of a shaft decoder. [15 marks]

7 Operating System Functions

(*a*) In the context of virtual memory management:

(*i*) What is *demand paging*? How is it implemented? [4 marks]

(*ii*) What is meant by *temporal locality of reference*? [2 marks]

(*iii*) How does the assumption of temporal locality of reference influence page replacement decisions? Illustrate your answer by briefly describing an appropriate page replacement algorithm or algorithms. [3 marks]

(*iv*) What is meant by *spatial locality of reference*? [2 marks]

(*v*) In what ways does the assumption of spatial locality of reference influence the design of the virtual memory system? [3 marks]

(*b*) A student suggests that the virtual memory system should really deal with "objects" or "procedures" rather than with pages. Make arguments both for and against this suggestion. [4 and 2 marks respectively]

.

8 Computation Theory

(a) Define precisely what is meant by the following:

(i) $f(x_1, x_2, \ldots x_n)$ is a Primitive Recursive (PR) function of arity n. [5 marks]

(ii) $f(x_1, x_2, \ldots x_n)$ is a Total Recursive (TR) function of arity n. [3 marks]

(b) Ackermann's function is defined by the following recursive scheme:

$$\begin{aligned} f(0, y) &= S(y) = y + 1 \\ f(x+1, 0) &= f(x, 1) \\ f(x+1, y+1) &= f(x, f(x+1, y)) \end{aligned}$$

For fixed n define

$$g_n(y) = f(n, y).$$

Show that for all n, $y \in \mathbb{N}$,

$$g_{n+1}(y) = {g_n}^{(y+1)}(1),$$

where $h^{(k)}(z)$ is the result of k repeated applications of the function h to initial argument z. [4 marks]

(c) Hence or otherwise show that for all $n \in \mathbb{N}$, $g_n(y)$ is a PR function. [4 marks]

(d) Deduce that Ackermann's function $f(x, y)$ is a TR function. [3 marks]

(e) Is Ackermann's function PR? [1 mark]

9 Numerical Analysis I

(*a*) What is meant by a *symmetric positive definite matrix*? [3 marks]

(*b*) Verify that $\mathbf{A} = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}$ is positive definite. [4 marks]

(*c*) The Choleski factorisation $\mathbf{A} = \mathbf{LDL}^T$ is to be applied to the solution of $\mathbf{Ax} = \mathbf{b}$, where $\mathbf{b} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$. It is found that

$$\mathbf{L} = \begin{pmatrix} 1 & \\ \frac{1}{2} & 1 \end{pmatrix}, \quad \mathbf{D} = \begin{pmatrix} 2 & \\ & \frac{3}{2} \end{pmatrix}.$$

The next step in the method is to solve $\mathbf{Ly} = \mathbf{b}$ to get $\mathbf{y} = \begin{pmatrix} 1 \\ \frac{1}{2} \end{pmatrix}$. Form the upper triangular system of equations needed to complete the solution. [4 marks]

(*d*) Solve these equations. [2 marks]

(*e*) What is meant by the *order of convergence* of an iterative process? [1 mark]

(*f*) State the Newton–Raphson formula for solving $f(x) = 0$ for scalar x. What is the order of convergence of this method? [2 marks]

(*g*) This method is used to solve $f(x) = x^2 - 4 = 0$ using IEEE Double Precision with a certain starting value x_0. It is found that the third iterate $x_3 \simeq 2.0006$, and $x_4 \simeq 2.00000009$. Very roughly, how many significant decimal digits of accuracy would you expect in x_5? Explain your answer. [4 marks]

10 Computer Graphics and Image Processing

(*a*) Describe an algorithm to draw a straight line using only integer arithmetic. You may assume that the line is in the first octant, that the line starts and ends at integer co-ordinates, and that the function *setpixel*(x, y) turns on the pixel at location (x, y). [8 marks]

(*b*) Describe Douglas and Pücker's algorithm for removing superfluous points from a line chain. [10 marks]

(*c*) Under what circumstances would it be sensible to employ Douglas and Pücker's algorithm? [2 marks]

END OF PAPER